

# NOWL: 구조 왜곡과 의미 손실 없이 토픽맵을 RDF로 변환하는 방법

(A Converting Method from Topic Maps to RDFs without  
Structural Warp and Semantic Loss)

신 신 애 \*      정 동 원 \*\*      백 두 권 \*\*\*  
(Shinae Shin)      (Dongwon Jeong)      (Doo-Kwon Baik)

**요약** 웹의 급속한 발전과 함께 웹 정보 자원을 보다 정확하게 이해할 수 있는 시멘틱 웹에 대한 필요성이 증가하고 있다. 현재 이와 관련된 두 가지 표준이 존재하는데 하나는 ISO/IEC JTC 1 표준인 토픽맵(Topic map)이고, 다른 하나는 W3C 표준인 RDF(Resource description framework)이다. 그러나 시멘틱 웹은 토픽맵을 적용한 정보자원과 RDF를 적용한 정보자원을 모두 인식할 수 있어야 하므로 토픽맵과 RDF간의 상호운용성이 요구된다. 이를 위해 토픽맵과 RDF간에 상호운용성 문제를 해결하기 위한 여러 변환 방법들이 제안되어 왔다. 그러나 기존의 방법들은 의미의 손실, 복잡한 구조, 불필요한 노드의 추가 등과 같은 문제점을 지니고 있었다. 이 논문에서는 이러한 문제를 해결하기 위한 새로운 방법인 NOWL (NO structural Warp and semantics Loss)을 제안한다. 제안한 방법은 기존 방법에 비해 토픽맵 본래의 의미 구조를 유지하고 불필요한 노드의 생성을 제거하는 등 여러 가지 장점을 지닌다.

**키워드** : 토픽맵, RDF, 시멘틱 웹, 웹 메타데이터

**Abstract** Need for machine-understandable web (Semantic web) is increasing in order for users to exactly understand Web information resources and currently there are two main approaches to solve the problem. One is the Topic map developed by the ISO/IEC JTC 1 and the other is the RDF (Resource Description Framework), one of W3C standards. Semantic web supports all of the metadata of the Web information resources, thus the necessity of interoperability between the Topic map and the RDF is required. To address this issue, several conversion methods have been proposed. However, these methods have some problems such as loss of meanings, complicated structure, unnecessary nodes, etc. In this paper, a new method is proposed to resolve some parts of those problems. The method proposed is called NOWL (NO structural Warp and semantics Loss). NOWL method gives several contributions such as maintenance of the original a Topic map instance structure and elimination of the unnecessary nodes compared with the previous researches.

**Key words** : Topic Map, RDF, Semantic Web, Conversion, Web Metadata

## 1. 소개

현재 웹은 시간과 공간을 넘어선 정보 자원의 방대한 저장소 및 정보의 교환과 공유를 위한 수단으로 이용되고 있다. 그러나 지금까지 대부분의 웹 정보들은 브라우저 및 검색엔진 등을 통하여 텍스트 매칭 기반으로 한

기계적인 관독만이 가능할 뿐 정보들이 지니는 의미 기반으로 하여 기계가 이해할 수 있도록 웹 정보 생성이 어렵다[1]. 따라서 웹 상에서 빠르게 증가하고 있는 방대한 정보들을 지금과 같은 수동적인 방식으로 검색하고 분석하기에는 한계가 있다. 최근 이러한 문제를 해결하기 위한 대안으로 시멘틱 웹(Semantic web)이 등장하였다. 시멘틱 웹은 웹 자원을 기계가 이해할 수 있도록 웹 정보들에 대해 보다 명확한 의미를 정의하고 전달할 수 있는 지능화된 웹을 의미한다.

시멘틱 웹의 실현을 위하여 가장 주목 받고 있는 접근 방법이 웹에 포함된 정보 자원을 설명하는 메타데이터를 이용하는 방법이다[1]. 웹 문맥 속에 웹 자원을 이

\* 종신회원 : 한국전산원 정보화표준부 팀장  
sashin@nca.or.kr

\*\* 정 회 원 : 군산대학교 정보통계학과 교수  
djeong@kunsan.ac.kr

\*\*\* 종신회원 : 고려대학교 컴퓨터학과 교수  
baik@software.korea.ac.kr

논문접수 : 2005년 5월 4일

실사완료 : 2005년 8월 21일

해할 수 있는 데이터인 메타데이터를 추가 기술함으로써 웹 자원들에 대한 구조화된 서술을 가능하게 한다. 현재 웹 정보 자원의 메타데이터를 기술하는 가장 대표적인 선도 방법으로는 ISO/IEC JTC 1에서 개발한 국제 표준인 토픽맵(Topic map)[2]과 W3C에서 개발한 표준인 RDF(Resource description framework)가 있다 [1,3].

토픽맵은 정보 자원에 대한 고수준의 인덱스 집합을 생성하는 것을 목적으로 하고 RDF는 정보자원에 대한 구조화된 메타데이터 및 논리적 추론을 통해 시멘틱 웹의 개념을 지원하기 위하여 개발되었다[4]. 토픽맵과 RDF는 모두 웹 상에 있는 정보 자원을 의미 모델을 기반으로 표현 및 교환하기 위해 연구 개발되었고, XML로 인코딩되며 노드와 아크로 이루어진 그래프로 표현되는 등 많은 유사성을 지닌다[5]. 그러나 토픽맵과 RDF는 시멘틱 웹을 목적으로 하면서도 조금 다른 구조와 개념 그리고 용어 사용으로 인한 불일치 문제를 내포하고 있다. 이러한 문제를 해결하고 보다 포괄적이고 효과적인 시멘틱 웹 개발을 위해서는 두 의미 집합(토픽맵과 RDF)간의 상호교환 및 공유가 가능해야 한다. 이는 보다 다양한 웹 정보자원의 메타데이터 통합을 가능하게 하여 완전한 시멘틱 웹 실현에 기여할 수 있다.

토픽맵과 RDF 정보간의 상호교환 및 공유를 위해 토픽맵 모델과 RDF 모델 간 변환을 위한 연구가 최근 활발히 진행되고 있다[6]. 그러나 이러한 기존의 연구 결과들은 다음과 같은 몇 가지 문제점을 지닌다.

- 불필요한 노드(Blank nodes)의 생성으로 인한 변환 결과의 높은 구조적 복잡도
- 기존 의미 구조의 왜곡(Warp) 및 일부 의미 정보(Semantic relations)의 손실
- 목시적 의미 정보로 인한 의미 결여 및 의미 해석상의 비용 증가

이 논문에서는 구조적 왜곡 및 정보의 손실 없이 토픽맵을 RDF로 변환하는 새로운 방법을 제안한다. 기존 방법에서는 토픽맵의 *topic* 요소와 *association* 요소를 중심으로 의미를 도출하고, 토픽맵의 *role* 요소를 간과함에 따라 의미 정보의 손실을 유발하였다. 따라서 이 논문에서는 *role* 요소를 핵심 요소로 정의하고 손실되는 의미 정보를 *role* 요소를 통하여 추출할 수 있는 변환 모델을 제안한다. 이 논문에서 제안하는 방법을 NOWL(NO structural Warp and semantic Loss)이라고 명명한다.

## 2. 관련 연구

이 장에서는 토픽맵 및 RDF의 개념과 특성에 대하여 간략하게 언급하고 기존의 토픽맵을 RDF로 변환하는

방법에 대한 연구를 분석한다.

### 2.1 토픽맵과 RDF 소개

토픽맵은 웹 상에서 정보 자원을 기술하고 다루는 방법을 정의하며 인덱스 역할을 수행한다. 토픽맵은 정보 자원이 의미하는 *subject*를 포착하고 *subject*간 관계성을 정의한다[7]. 토픽맵은 n차 레이블 그래프(n-ary labeled graph)로서 노드와 아크로 구성되어 있다[5]. 그림 1은 토픽맵의 기본적인 스키마를 보여준다.

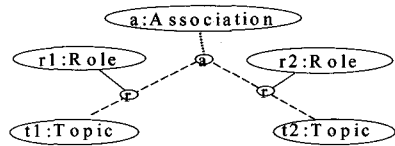


그림 1 기본적인 토픽맵 스키마

정보자원은 *topic*을 통하여 해당 정보자원의 주제를 정의하며 *topic*간의 관계는 *association*으로 정의한다. 이때 *topic*들은 *role*을 통하여 *association* 관계에서의 역할을 표현하게 된다.

RDF는 메타데이터를 구성하여 웹 응용들간 정의에 대한 의미를 상호 이해할 수 있도록 하기 위하여 W3C에 의해 개발되었다[1]. 그림 2는 기본적인 RDF 스키마를 보여준다.

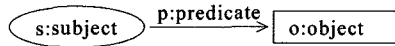


그림 2 기본적인 RDF 스키마

RDF 스키마(RDF Scheme)은 *subject*, *predicate* 및 *object*의 집합으로 구성된다.

**정의 1. (RDF 스키마)** S, P, O를 각각 *subject*, *predicate*, *object*의 집합이라고 가정했을 때, RDF 스키마 R은 다음과 같이 3-튜플로 표현된다.

$$R = (S, P, O)$$

### 2.2 토픽맵-to-RDF 변환 방법

전역적인 시멘틱 웹의 활용성 증대를 위하여 다양한 토픽맵과 RDF간 변환 방법이 제안되었다[4,5,8,9]. [4]는 토픽맵 모델을 RDF로 표현하는 방법을 제시하면서 토픽맵과 RDF간의 의미 교환을 가능하도록 하기 위해서 이들간의 사상을 위한 새로운 어휘를 생성하였다. 사전에 사상 어휘(Mapping vocabularies)를 사용하여 토픽맵 모델을 정의함에 따라 사용자의 의도를 반영한 변환이 가능하나 새로운 매핑 어휘의 사용으로 기존 토픽맵 인스턴스들을 사상 어휘에 따라 재작성해야 하고 사용자가 새로운 어휘를 이해해야 하는 단점을 지닌다.

[5]는 토픽맵의 노드를 t-node(*topic node*)와 a-node

(association node)로 분류하고 t-node를 RDF의 resource로 a-node를 RDF의 predicate로 변환하는 노드 유형에 기반한 변환 방법을 제안하였다. 그러나 role 요소를 별도의 노드 유형으로 식별하지 않음으로써 role을 통하여 추출될 수 있는 정보의 손실을 초래하며 토픽맵의 association을 중심으로 RDF로의 변환을 수행함으로써 변환 이후에 의미에 대한 변형을 유발한다.

[8]은 RDF의 기본구조를 토픽맵으로 변환하는 방법과 그 역 변환 방법을 제안하였다. topic과 role간의 관계를 RDF로 정의하고 정의된 RDF간의 관계를 다시 생성하며, 관계 정의 시 공백 노드(Blank node)를 추가함으로써 구조적으로 복잡한 변환 결과를 생성한다.

[9]는 XML 토픽맵과 RDF간 변환을 위하여 XTM2RDF 변환기를 제시하였다. 이 방법은 변환을 위해 새로운 어휘를 사용함으로써 기존의 토픽맵 데이터를 그대로 사용하지 못하는 문제를 지닌다. 아울러 role 요소를 간과함으로써 role을 통해 도출될 수 있는 의미 정보를 상실하게 된다.

### 3. NOWL 모델

이 장에서는 이 논문에서 제안하는 새로운 변환 방법인 NOWL(NO structural Warp and semantics Loss) 모델의 개념을 소개하고 그 구조 및 알고리즘 등에 대하여 기술한다.

#### 3.1 NOWL 접근방식의 기본 개념

토픽맵과 RDF는 유사한 개념과 동일한 목적을 지니지만 서로 다른 스킴과 구문을 지닌다. RDF 그래프는 subject와 object를 나타내는 두 개의 노드와 이들을 연결하는 하나의 아크로 구성된 이진 모델이며, subject와 object간 관계는 방향성을 가진다. 반면 토픽맵 그래프는 topic, role, association이 각각 다수개의 노드로 표현되고 이들 노드간의 관계는 다수의 아크로 정의되는 n-ary 그래프 모델이다. 토픽맵은 RDF와는 달리 관계에 방향성이 없다.

토픽맵을 RDF로 변환하는데 있어 가장 중요한 문제는 ① 구조의 변형이나 복잡도의 증가를 배제해야 한다는 점과 ② 소스 모델에서 지니는 원래의 의미 손실 및 의미의 구조적 왜곡이 없어야 한다는 점이다. 그러나 2.2절에 설명한 기존의 대표적인 변환 방법 중 [4]와 [9]는 새로운 어휘를 사용하고, [5]와 [8]은 각각 ①과 ②의 문제를 내포하고 있다.

그림 3과 같은 샘플 토픽맵 그래프를 가정해 보자. 그림 3은 토픽맵과 RDF간의 변환을 위하여 기존 연구들에서 많이 사용하고 있는 토픽맵 기본 샘플이다.

주어진 토픽맵을 [8]의 방법을 통해 RDF로 표현하면 그림 4와 같다. 그림 4에서, 소스 모델, 즉 토픽맵 인스

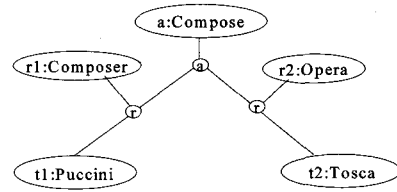


그림 3 토픽맵 기본샘플 예제

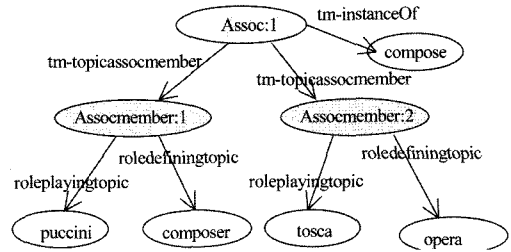


그림 4 [8]에 의해 변환된 RDF 그래프

턴스에는 포함되어 있지 않은 {Assoc:1, Assocmember:1, Assocmember:2}와 같은 추가적인 노드들이 생성되며 이는 전체적인 구조의 복잡도를 증가시키는 결과를 초래하게 된다. 또한 추가 노드를 명명함에 있어 명확하지 않은 임의의 명칭을 사용함에 따라 친밀성 결여와 관계에 대한 이해를 어렵게 한다.

그림 3의 토픽맵 인스턴스를 [5]의 변환 모델을 이용할 경우, 변환 RDF 그래프는 그림 5와 같다. [5]에서 제안한 변환 모델은 [8]의 변환 모델과는 달리 변환 결과에서 추가적인 노드를 생성하지 않으나 정보에 대한 손실이 발생하고 본래의 의미 구조와는 초점이 다른 의미 구조로 변환되어 정보의 손실과 의미 전달에 변형을 발생시킨다.

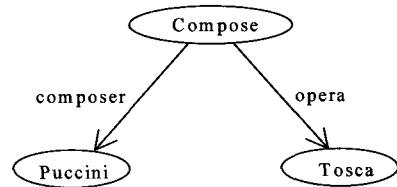





그림 5 [5]에 의해 변환된 RDF 그래프

#### 3.2 NOWL 변환 모델

이 논문에서 사용한 표기 및 기호와 그 의미는 표 1과 같다.

NOWL 모델은 입력(토픽맵), 출력(RDF) 및 변환 함수로 구성된다. NOWL에서의 변환 함수  $Mf$ 는 ① subject와 object를 식별하는 부분과 ② predicate를 식별하는 부분 그리고 ③ 최종적으로 완전한 RDF 인스턴

표 1 주요 표기 및 기호

$TM_i$	토픽맵 인스턴스를 의미하며, 토픽맵 그래프와 동일한 의미로 사용됨	$O_i$	임의의 RDF 인스턴스를 구성하는 <i>object</i> 들 중의 하나로 <i>object</i> 인스턴스를 의미
$RDF_i$	RDF 인스턴스를 의미하며, RDF 그래프와 동일한 의미로 사용됨	$P_i$	임의의 RDF 인스턴스에서 <i>subject</i> 와 <i>object</i> 간의 관계를 표현하는 <i>predicate</i> 의 인스턴스
$t_i$	임의의 토픽맵 인스턴스를 구성하는 <i>topic</i> 들 중의 하나로 실제 <i>topic</i> 인스턴스를 의미	$newP$	임의의 RDF 인스턴스에서 <i>subject</i> 와 <i>object</i> 간의 관계를 표현하기 위해서 새로 생성한 <i>predicate</i> 인스턴스
$a_i$	임의의 토픽맵 인스턴스에서 <i>topic</i> 간의 관계를 정의하는 <i>association</i> 요소		RDF 또는 토픽맵 그래프에서 노드에 해당하는 것으로, <i>topic</i> , <i>role</i> , <i>association</i> , <i>subject</i> , <i>object</i> 등을 나타냄
$r_i$	임의의 토픽맵 인스턴스에서 <i>topic</i> 이 <i>association</i> 에서 가지는 역할로 실제 $i^{th}$ <i>role</i> 인스턴스를 의미		임의의 토픽맵 인스턴스의 그래프에서 노드간 관계에 해당하는 <i>arc</i> 를 표현
$S_i$	임의의 RDF 인스턴스를 구성하는 <i>subject</i> 들 중의 하나로 실제 <i>subject</i> 인스턴스를 의미		주어진 토픽맵 인스턴스에서 추출된 RDF로 변환될 <i>arc</i> 를 표현하며 이들의 집합이 RDF 인스턴스를 구성함

스를 생성하기 위한 예외 처리 부분으로 구성되어 있다.

**정의 2. (NOWL: NOWL 변환 모델)** 입력 값인 토픽맵 인스턴스의 집합인  $TM$ 을  $I$ , 변환 결과인 RDF 인스턴스의 집합을  $O$ , 그리고 변환 함수를  $Mf$ 라 하고 변환 함수의 세 부분인 ①, ②, ③을 각각  $IE$ ,  $SP$ ,  $AE$ 라고 했을 때, NOWL는 다음과 같이 정의된다.

**NOWL ( $I$ ,  $O$ ,  $Mf$ )**

- $I = \{TM_i \mid i = \text{전체 토픽맵 인스턴스 중 임의의 인스턴스}\} = \sum TM_i = TM$
- $O = \{RDF_j \mid j = \text{전체 RDF 인스턴스 중 임의의 인스턴스}\} = \sum RDF_j = RDF$
- $Mf = (IE, SP, AE)$

정의 2에서, 변환 함수  $Mf$ 는 세 개의 부분 변환 함수 ( $IE$ ,  $SP$ ,  $AE$ )으로 구성되며 이들 각 부분 변환 함수는 순차적, 단계적으로 수행되며 입력, 출력 및 적절한 기능 수행을 위한 규칙들을 지닌다.  $IE$ ,  $SP$ ,  $AE$ 에 대한 구조는 각각 정의 3, 4, 5와 같다.

**정의 3. ( $IE$ : *subject*와 *object* 식별 부분 변환 함수)** 입력 값인 토픽맵 인스턴스로 부터 *subject*와 *object*에 해당하는 요소를 추출하여 *subject*와 *object* 쌍의 집합인  $(S, O)$ 를 생성한다.

**$IE (IN_{IE}, OUT_{IE}, \Gamma_{IE})$**

- $IN_{IE} = TM = \sum TM_i = \{TM_i \mid TM_i \in TM_{all}, i = \# \text{ of Topic map instances}\}$   
-  $TM_{all}$ : 모든 토픽맵 인스턴스
- $OUT_{IE} = (S, O) = \sum (S_i, O_i) = \{(S_i, O_i) \mid S_i \in S, O_i \in O\}$
- $\Gamma_{IE}$ : 세부적인 식별 규칙들의 집합

정의 4는  $IE$ 의 결과인 *subject*와 *object*들간의 관계, 즉 *predicate*을 정의하고 *subject*, *predicate* 및 *object*로 구성된 RDF 문장(RDF statement) 집합( $S, P, O$ )를 생성하게 된다. *predicate*를 정의하여 RDF statement를 생성하는 함수를  $SP$ 라고 할 때 다음과 같다.

**정의 4. ( $SP$ : *predicate* 정의/생성하는 함수)** 정의 3에서 추출된 *subject*와 *object*간의 관계를 정의함으로써 RDF 문장 집합인  $(S, P, O)$ 를 생성하는 부분 변환 함수이다. 정의 3의 부분 변환 함수  $IE$ 의 결과인  $OUT_{IE}$ 를  $SP$  부분 변환 함수의 입력  $IN_{SP}$ 라고 정의하고 *predicate*를 생성하는 규칙 집합과 변환된 결과인 RDF 문장을 각각  $\Gamma_{SP}$ 와  $OUT_{SP}$ 로 정의한다.

**$SP (IN_{SP}, OUT_{SP}, \Gamma_{SP})$**

- $IN_{SP} = OUT_{IE} = (S, O) = \sum (S_i, O_i) = \{(S_i, O_i) \mid S_i \in S, O_i \in O\}$
- $OUT_{SP} = (S, P, O) = \sum (S_i, P_i, O_i) = \{(S_i, P_i, O_i) \mid S_i \in S, P_i \in P, O_i \in O\}$
- $\Gamma_{SP}$ : *predicate* 정의 및 RDF 문장 생성 규칙들의 집합

마지막으로, 완전한 RDF 모델로의 변환을 위해서는 예외 처리 규칙의 적용이 요구되며 이 연산을 수행하는 함수를  $AE$ 이라고 할 때,  $AE$ 는 다음과 같이 정의된다.

**정의 5. ( $AE$ : 예외 처리 부분 변환 함수)** RDF 특성인 정보자원에 대한 접근성에 맞는 결과를 생성한다.

**$AE (IN_{AE}, OUT_{AE}, \Gamma_{AE})$**

- $IN_{AE} = OUT_{SP} = (S, P, O) = \sum (S_i, P_i, O_i) = \{(S_i, P_i, O_i) \mid S_i \in S, P_i \in P, O_i \in O\}$
- $OUT_{AE} = \sum (S_i, P_i, O_i)$
- $\Gamma_{AE}$ : 세부적인 예외 처리 규칙 집합

**3.3 변환 규칙**

이 절에서는 보다 상세한 변환 규칙과 과정에 대하여 기술한다. 앞서 기술하였듯이, NOWL 변환 모델은 토픽맵 인스턴스를 입력 받아 변환 함수인  $Mf$ 를 통해 RDF 인스턴스를 생성한다.  $Mf$ 는 부분 변환 함수인  $IE$ ,  $SP$ ,  $AE$ 로 구성되어 있으며 순차적인 연산 순서를 지닌다. 따라서 변환 알고리즘은 크게 3 단계로 이루어지며 전체적인 알고리즘은 다음과 같다.

# FUNCTION:  $Mf$

```

# IN:  $\Sigma TM_i$ 
# OUT:  $\Sigma RDF_i$ 
START_FUNCTION {
  IE (IN:  $\Sigma TM_i$ , OUT:  $\Sigma (S_i, O_i)$ ) {
    Select topics and roles from  $TM_i$ ;
    Convert topics and roles to subjects and
    objects;
  }
  SP (IN:  $\Sigma TM_i$ ,  $\Sigma (S_i, O_i)$ , OUT:  $\Sigma (S_i, P_i, O_i)$ ) {
    Convert associations as predicates;
    Convert roles as predicates;
    Create new predicates;
    Remove duplicated predicates;
  }
  AE (IN:  $\Sigma (S_i, P_i, O_i)$ ,  $\Sigma RDF_i$ ) {
    Remove triples where subjects are
    non-addressable concepts.
  }
} END_FUNCTION

```

위 전체 변환 알고리즘에서, 변환을 위해 이용되는 각 규칙들을 정의하면 규칙 1, 규칙 2, 규칙 3, 규칙 4, 규칙 5와 같이 정의된다.

**규칙 1. ( $\Gamma_{IE}$ : subject와 object 식별 규칙)** 토픽맵 인스턴스  $TM_i$ 에서 topic과 role을 추출하여 RDF 모델의 subject와 object로 사상 시킨다.

```

r1.1a:  $(t_i, S_i) \rightarrow (t_j, O_i) /* t_i \text{ has an association}
\text{relationship with } t_j*/$ 
r1.1b:  $(t_j, O_i) \rightarrow \text{createPair}(t_i, t_j) \equiv \text{createPair}(s_i, o_i)$ 
r1.2a:  $(t_j, S_j) \rightarrow (t_i, O_j) /* t_i \text{ has an association}
\text{relationship with } t_j*/$ 
r1.2b:  $(t_i, O_j) \rightarrow \text{createPair}(t_j, t_i) \equiv \text{createPair}(s_j, o_j)$ 
r1.3a:  $(r_i, S_i) \rightarrow (r_j, O_i) /* r_i \text{ and } r_j \text{ are connected}
\text{with a same association}*/$ 
r1.3b:  $(r_j, O_i) \rightarrow \text{createPair}(r_i, r_j) \equiv \text{createPair}(s_i, o_i)$ 
r1.4a:  $(t_i, S_i) \rightarrow (r_i, O_i) /* r_i \text{ is a role of } t_i */$ 
r1.4b:  $(r_i, O_i) \rightarrow \text{createPair}(t_i, r_i) \equiv \text{createPair}(s_i, o_i)$ 
r1.5a:  $(t_j, S_j) \rightarrow (r_j, O_j) /* r_j \text{ is a role of } t_j */$ 
r1.5b:  $(r_j, O_j) \rightarrow \text{createPair}(t_j, r_j) \equiv \text{createPair}(s_j, o_j)$ 

```

규칙 1에 의해서 주어진 토픽맵 인스턴스로부터 subject와 object 쌍의 집합  $\Sigma(S, O)$ 를 얻으면 다음은 predicate을 정의하게 된다. 이는 정의 4의 SP 함수의 규칙 집합인  $\Gamma_{SP}$ 를 통해 이루어진다.

**규칙 2. ( $\Gamma_{SP}$ : association을 predicate으로 정의하는 규칙)** 토픽맵에서 association은 topic간 또는 role간의 관계로 정의될 수 있다.

```

r2.1a:  $(t_i, S_i) \wedge (t_i, O_i) \rightarrow (a, P_i) /* a_i \text{ is an}$ 
```

```

association between  $t_i$  and  $t_j*/$ 

```

```

r2.1b:  $(a, P_i) \rightarrow \text{createTriple}(t_i, a, t_j) \equiv \text{createTriple}
(s_i, p_i, o_i)$ 

```

```

r2.2a:  $(r_i, S_j) \rightarrow (r_j, O_j) \rightarrow (a, P_j) /* r_i \text{ and } r_j \text{ are}
\text{connected with a same association } a_i*/$ 

```

```

r2.2b:  $(a, P_j) \rightarrow \text{createTriple}(r_i, a, r_j) \equiv \text{createTriple}
(s_j, p_j, o_j)$ 

```

role도 topic간의 관계를 나타내는 predicate으로 변환될 수 있다. 이 때, topic간의 관계의 방향성을 고려해야 한다.  $t_1$ 이 subject가 되고  $t_2$ 가 object가 되면  $r_2$ 가 predicate이 되고,  $t_2$ 가 subject가 되고  $t_1$ 이 object가 되면  $r_1$ 이 predicate이 된다. 이를 정의하면 규칙 3과 같다.

**규칙 3. ( $\Gamma_{SP2}$ : role을 predicate으로의 정의 규칙)** 토픽맵에서 role은 topic간 관계로도 정의될 수 있다.

```

r3.1a:  $(t_i, S_i) \wedge (t_j, O_i) \rightarrow (r_j, P_i) /* r_j \text{ is a role of } t_j*/$ 
r3.1b:  $(r_j, P_i) \rightarrow \text{createTriple}(t_i, r_j, t_j) \equiv \text{createTriple}
(s_i, p_i, o_i)$ 

```

```

r3.2a:  $(t_j, S_j) \wedge (t_i, O_j) \rightarrow (r_i, P_j) /* r_i \text{ is a role of } t_i*/$ 
r3.2b:  $(r_i, P_j) \rightarrow \text{createTriple}(t_j, r_i, t_i) \equiv \text{createTriple}
(s_j, p_j, o_j)$ 

```

topic과 role이 각각 RDF 모델의 subject와 object로 변환될 수 있으며 이들간에는 새로운 predicate을 정의하여야 한다. 규칙 4는 이 연산을 위한 규칙을 정의한다.

**규칙 4. ( $\Gamma_{SP3}$ : 새로운 predicate 정의 규칙)** topic과 role 간의 관계를 정의하기 위한 새로운 predicate을 생성한다. 즉,  $t_i$ 와  $r_i$ 에 해당하는  $t_i$  추출된 subject와 object를  $S_i$ 와  $O_i$ 라고 할 때 새로운 predicate  $newP'$ 이 생성된다.

```

R4.1a:  $(t_i, S_i) \wedge (r_i, O_i) \rightarrow (newP', P_i) /*r_i \text{ is a role}
\text{of } t_i */$ 

```

```

R4.1b:  $(newP', P_i) \rightarrow \text{createTriple}(t_i, newP', r_i)$ 

```

```

R4.2a:  $(t_j, S_j) \wedge (r_j, O_j) \rightarrow (newP'', P_j) /*r_j \text{ is a role}
\text{of } t_j */$ 

```

```

R4.2b:  $(newP'', P_j) \rightarrow \text{createTriple}(t_j, newP'', r_j)$ 

```

지금까지 기술한 규칙들을 기반으로 변환 연산이 수행될 경우 동일한 topic간에 중복된 관계, 즉 중복된 predicate이 생성될 수 있다. 규칙 2와 규칙 3에서 동일한  $(S, O)$ 인  $(t_i, t_j)$ 에 대해서 두개의 중복된 predicate이 생성된다. 규칙 2에서는  $(t_i, a, t_j)$ 를 규칙 3에서는  $(t_i, r_j, t_j)$ 를 생성하는 것이다. 이것은 RDF의 특성인 subject와 object간에는 하나의 predicate만을 허용하는 특성에 위배된다. 이 문제는 topic이 role의 인스턴스에 해당한다는 topic과 role의 관계를 통해서 해결될 수 있다. 따라서 이 문제는 role들간의 관계에만 association

을 *predicate*로 변환되도록 하고 *topic*들간의 관계는 *association*과 *role*을 합성한 *c-predicate* (*Composite predicate*)을 생성함으로써 해결할 수 있다.

**규칙 5. ( $\Gamma_{SP4}$ : 정제 규칙)** 규칙 2와 3에서 발생하는 중복성 문제는 우선 *topic*  $t_i, t_j$ 간 *association*  $a_j$ 로 생성된 RDF 문장인  $(t_i, a_j, t_j)$ 는 제거하고,  $t_i, t_j$ 에는 *association*과 *role*을 합성한 *c-predicate*을 생성한다.

R5.1:  $(S_i, P_i, O_i) \rightarrow (t_i, a, t_j) \wedge (t_j, a, t_i) /* t_i \text{ and } t_j \text{ are elements of } (S, O) /*$

R5.2:  $(t_i, a, t_j) \wedge (t_j, a, t_i) \rightarrow \text{removeTriple}(t_i, a, t_j) \wedge \text{removeTriple}(t_j, a, t_i)$

R5.3:  $\text{removeTriple}(t_i, a, t_j) \wedge \text{removeTriple}(t_j, a, t_i) \rightarrow \text{createTriple}(t_i, a_r, t_j) \wedge \text{createTriple}(t_j, a_r, t_i) /* a_r, a_r \text{ is a composite predicate of association and role } /*$

**4. 변환 예제**

이 장에서는 샘플 데이터를 통하여 NOWL 변환 알고리즘을 이용한 변환 결과를 기술한다.

**4.1 예제 1: 토피맵 기본 모델**

이 논문에서 기본 모델은 토피맵의 핵심 요소인 *topic, association, role*로 구성된 단일 토피맵을 의미하며 기존의 많은 관련 연구들에서 공통적으로 활용되고 있는 예제(그림 3)를 이용한다.

**1) Subject와 Object 식별**

그림 3의 토피맵 예제에 규칙 1을 적용하여 *topic*과 *role*을 통하여 식별된 *subject*와 *object*는 그림 6과 같으며 나열한 결과는 다음과 같다.

$(S, O) = \{ (Puccini, Tosca), (Tosca, Puccini), (Puccini, Composer), (Tosca, Opera), (Composer, Opera) \}$

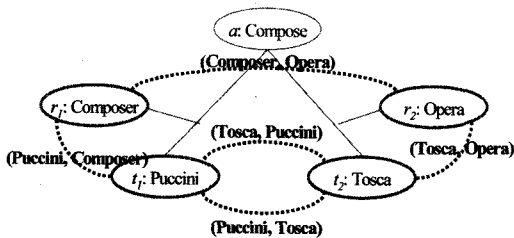


그림 6 식별된 subject와 object

**2) Predicate 정의 및 정제화**

규칙 1에 의해 생성된 그림 6의 *subject*와 *object* 쌍의 집합  $(S, O)$ 에 규칙 2, 규칙 3, 규칙 4를 적용하여 처리한 결과는 다음과 같다.

$M_{11} = (Puccini, Compose, Tosca)$

$M_{12} = (Puccini, Opera, Tosca)$

$M_{21} = (Tosca, Compose, Puccini)$

$M_{22} = (Tosca, Composer, Puccini)$

$M_3 = (Puccini, is-a, Composer)$

$M_4 = (Tosca, is-a, Opera)$

$M_5 = (Composer, Compose, Opera)$

위 결과에서,  $M_{11}, M_{12}$ 와  $M_{21}, M_{22}$ 는 규칙 5에 의해 *c-predicate*으로 변환된다. 즉,  $M_{11}, M_{12}, M_{21}, M_{22}$ 는 삭제되고 각각 *c-predicate*으로  $(Puccini, Compose.Opera, Tosca)$ 와  $(Tosca, Compose.Composer, Puccini)$ 가 생성된다. 따라서 최종적으로 그림 7과 같은 결과를 얻게 되며 나열하면 다음과 같다.

$M_1 = (Puccini, Compose.Opera, Tosca)$

$M_2 = (Tosca, Compose.Composer, Puccini)$

$M_3 = (Puccini, is-a, Composer)$

$M_4 = (Tosca, is-a, Opera)$

$M_5 = (Composer, Compose, Opera)$

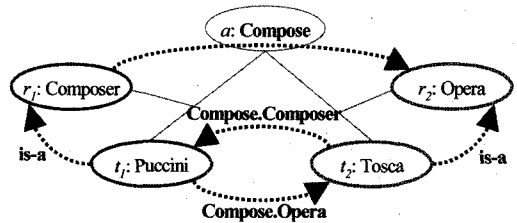


그림 7 predicate 정의 및 정제 결과

토피맵의 *topic, association, role*은 'thing'을 표현하는데 'thing'은 URI를 가지고 있는 주소 지정이 가능한 자원(Addressable resources)이 될 수도 있고, 단지리터럴(Literal, string)인 주소 지정이 불가능한 개념일 수도 있다[10]. 그러나 RDF에서의 자원은 반드시 URI를 가지는 주소 지정이 가능한 자원이어야 한다 [1]. 따라서 RDF에서는 *subject*가 자원을 나타내므로 토피맵을 RDF로 변환 시 주소 지정이 가능한 *topic*과 *role*만이 *subject*가 될 수 있다. 따라서 완전한 RDF로의 변환을 위해서는 주소 지정이 불가능한 개념은 *subject*로 변환될 수 없도록 하는 규칙 적용이 요구되며 이를 위한 규칙을 정의하면 다음과 같다.

**규칙 6. ( $\Gamma_{AE}$ : 예외 처리 규칙)** AE 함수에 대한 예외처리 규칙 집합  $\Gamma_{AE}$ 는 다음과 같다.

R6.1:  $(S_i, P_i, O_i) \rightarrow \text{isLiteral}(S_i)$

R6.2:  $\text{isLiteral}(S_i) \rightarrow \text{removeTriple}(S_i, P_i, O_i)$

**4.2 예제 2: 록 밴드 The Clash에 대한 샘플 Topic map**

**1) 샘플 개요**

이 절에서는 4.1절에서 사용한 토픽맵 보다 다소 복잡한 토픽맵 샘플을 대상으로 NOWL을 이용하여 변환된 결과를 기술한다. 그림 8은 록 밴드 The Clash에 대한 샘플 토픽 맵 그래프로서, 토픽맵 전문 사이트인 Techquila의 XML로 인코딩 되어 있는 샘플을 그래프로 표현한 것이다[11].

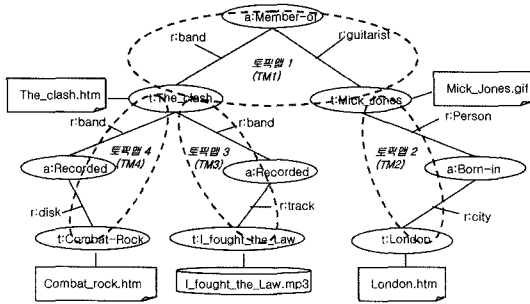


그림 8 록 밴드 The Clash에 대한 샘플 토픽맵[16]

그림 8은 다음과 같이 4개의 토픽맵들로 구성되어 있다.

- TM<sub>1</sub> : The Clash는 Mick Jones를 구성원으로 (member-of)하며, The Clash는 band이고 Mick Jones는 guitarist 이다.
- TM<sub>2</sub> : Mick Jones는 London에서 태어났으며 (born-in), Mick Jones는 person이고 London은 city 이다.
- TM<sub>3</sub> : The Clash는 I fought the law를 녹음하였으며 (recorded), The Clash는 band이고 I fought the law는 track이다.
- TM<sub>4</sub> : The Clash는 The Combat Rock을 녹음하였으며(recorded), The Clash는 band이고 The Combat Rock은 disk이다.

2) Subject와 Object 식별

그림 8의 샘플에 규칙 1을 적용하여 식별된 subject와 object의 쌍 (S, O)는 그림 9와 같다.

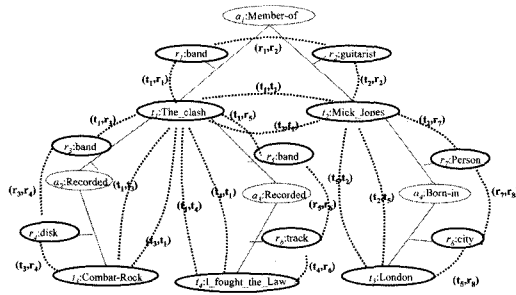


그림 9 규칙 1을 적용하여 식별한 subject와 object

3) Predicate 정의 및 정제화

그림 9와 같이 규칙 1에 의해 생성된 subject와 object 쌍의 집합 (S, O)에 predicate을 정의하는 규칙 2에서부터 규칙 5를 적용하면 다음과 같은 결과가 도출된다.

- $M_1 = (The\_Clash, Member\_of.Guitarist, Mick\_Jones)$ ,
- $M_2 = (Mick\_Jones, Member\_of.Band, The\_Clash)$
- $M_3 = (Band, Member\_of, Guitarist)$ ,  $M_4 = (The\_Clash, is-a, Band)$ ,
- $M_5 = (Mick\_Jones, is-a, Guitarist)$
- $M_6 = (The\_Clash, Recorded.Disk, Combat\_Rock)$ ,  $M_7 = (Combat\_Rock, Recorded.Band, The\_Clash)$ ,
- $M_8 = (The\_Clash, is-a, Band)$ ,  $M_9 = (Combat\_Rock, is-a, Disk)$ ,
- $M_{10} = (Band, Recorded, Disk)$
- $M_{11} = (The\_Clash, Recorded.Track, I\_fought\_the\_Law)$ ,
- $M_{12} = (I\_fought\_the\_Law, Recorded.Band, The\_Clash)$
- $M_{13} = (The\_Clash, is-a, Band)$ ,  $M_{14} = (I\_fought\_the\_Law, is-a, Track)$ ,
- $M_{15} = (Band, Recorded, Track)$
- $M_{16} = (Mick\_Jones, Born-in.City, London)$ ,  $M_{17} = (London, Born\_in.Person, Mick\_jones)$
- $M_{18} = (Mick\_Jones, is-a, Person)$ ,  $M_{19} = (London, is-a, City)$ ,
- $M_{20} = (Person, Born\_in, City)$

도출된 결과를 보면,  $M_4, M_8, M_{13}$ 이 중복됨을 알 수 있다. 이는 topic The\_Clash가 토픽맵1, 토픽맵3, 토픽맵4에서 동일한 role인 r:band를 중복하여 정의하고 있기 때문이다. 따라서 중복을 제거하고  $M_4, M_8, M_{13}$  중 하나만을 남김으로써 4개의 토픽맵에서 총 18개의 RDF 문장이 생성된다.

5. 비교 평가

이 장에서는 기존 변환 모델과 이 논문에서 제안하는 NOWL 변환 모델간의 비교 평가 결과에 대하여 기술한다.

5.1 평가 모델

이 논문에서는 NOWL 모델이 기존 모델의 주요 문제점이었던 변환 후 의미의 손실과 변환 결과의 구조적인 복잡도를 해결함을 증명하기 위해 의미 손실을, 변환된 RDF 인스턴스의 구조적인 복잡도, 질의 효율성 등의 평가 항목을 이용한다.

선정한 평가 항목은 [6]에서 기존의 변환 방법들을 비교하기 위해 이용되었으며 이 문에서도 이를 기반으로 평가를 수행한다. [16]에서 기술하고 있는 비교 연구는 W3C에서 수행한 연구 결과로서 기존 변환 방법들을 제안한 저자들이 주축이 되어 수행하였다. 이 논문에서 이용하고 있는 예제는 [6]에서 여러 변환 방법들을 비교하기 위해 사용된 예제 중 일부이며 평가 항목 또한 이 연구 내용에 따른 것이다. 따라서 이 논문에서 제시하고 있는 평가 모델 및 평가 결과에 대한 타당성을 제공한다.

### (1) 의미 손실률 비교 평가 모델

의미 손실률은 주어진 토픽맵 모델이 지니는 의미 정보와 변환된 RDF 모델에서 표현하고 있는 의미 정보간의 차이를 의미한다. [6]에서는 변환된 RDF의 개수를 통해 변환 방법들의 의미 손실률을 비교하였다. 이 논문에서는 손실된 개수가 아닌 손실률이 몇 %에 이르는지를 보이기 위해 다음과 같은 계산 모델 (E1)을 이용한다. 이는 단순히 보여지는 결과의 형태가 다를 뿐 그 기본적인 평가 개념은 [6]의 평가 모델을 따르고 있다. 토픽맵 모델의 의미 개수를  $n(TM)$ 이라 하고 변환된 RDF 모델에서 추출 가능한 모든 의미의 개수를  $n(RDF)$ 라고 할 때 손실된 개수,  $n(Loss)$ 는  $(n(TM) - n(RDF))$ 이다.  $originM_i$ 와  $convertedM_j$ 를 각각 토픽맵과 RDF 모델에서 추출한 하나의 의미 정보라고 했을 때, 의미 손실률  $LossR$ 은 다음과 같다.

$$LossR(CM) = \frac{\sum originM_i - \sum convertedM_j}{\sum originM_i} \\ = \frac{n(TM) - n(RDF)}{n(TM)} = \frac{n(Loss)}{n(TM)}$$

### (2) 구조 복잡도 비교 평가 모델

구조적 복잡도는 불필요한 노드의 생성 등으로 인하여 변환된 RDF 인스턴스 구조가 복잡해지게 되는 문제점을 다룬다. [6]에서는 변환된 RDF 결과의 이해의 편의성/가독성과 다른 모델로의 변환 용이성 문제를 자연성(Naturalness)이라고 정의하고 이를 평가 항목으로 이용하였다. 즉 변환 결과에 새로운 노드의 추가 및 이에 따른 아크의 추가, 새로운 어휘의 추가에 따른 친밀성 등을 비교 연구하였다.

이 논문에서는 새로운 노드와 아크의 추가로 인한 문제점을 구조 복잡도로서 정의한다. [6]의 비교 평가의 경우, 변환된 RDF의 노드와 아크의 개수를 통해 비교 연구를 수행하였으나 2진 트리로 생성되는 RDF의 특성상 두 대상의 평가 결과는 동일하다. 이 논문에서는 생성된 RDF의 노드 개수를 대상으로 평가 모델을 정의한다.

구조적 복잡도의 값을 계산하기 위하여 특정 모델의 노드 개수를 기준으로 다른 모델을 비교한다. 이 논문에서 기준 모델을 *PivotCM*이라고 정의하고 계산 결과인 복잡도를  $Complex(CM)$ 이라고 정의한다. 기준 *PivotCM*이 있고, 임의의 변환 모델의 복잡도  $Complex(CM)$ 이 1.5 인 경우 변환 모델의 복잡도가 기준 모델보다 1.5배 높다는 것을 의미한다.

이 논문에서는 *PivotCM*을 비교대상인 기준 모델 [8]로 정의하고 NOWL 변환 모델의 복잡도를 계산한다. [5]의 변환 모델은 의미 손실을 유발하므로 복잡도 비교가 무의미하다. 따라서 [5]의 변환 모델은 평가 대상에

서 제외한다. *PivotCM*인 [8]의 변환 모델과 NOWL 모델에 의해 변환된 RDF 인스턴스들이 지니는 노드를 각각  $n(PivotN_i)$ ,  $n(NOWL N_i)$ , 그리고 추출된 전체 노드의 개수를 각각  $n(Pivot)$ ,  $n(NOWL)$ 라고 했을 때, 구조적 복잡도  $Complex(CM)$ 은 다음과 같다. 여기에서  $cmN_i$ 는 비교 대상으로서 주어진 변환 모델에 의해 생성된 RDF 인스턴스로서 현재 비교 대상은 NOWL 변환 모델이 된다.

### (3) 질의 효율성을 위한 평가 모델

변환된 RDF 인스턴스는 질의를 통해 효율성을 확인할 수 있다. 질의 효율성은 구조적 복잡도 및 의미 손실 문제와 깊은 연관성을 지닌다. 즉 구조적인 복잡도가 증가함으로써 주어진 질의에 대한 결과를 생성하는 프로세스/알고리즘이 복잡해지고 또한 최종 결과를 검색하는데 많은 시간이 소요된다. 이 논문에서는 주어진 임의의 질의 결과를 생성하기 위해 요구되는 시간을 통해 각 제안 모델의 질의처리 시간, 즉 질의 효율성을 평가한다.

변환된 RDF에 대한 질의처리 시간은 시스템 특성, 구현된 검색 알고리즘 및 인덱스 구조 등에 따라 달라질 수 있으나 이 논문에서는 변환 모델간의 특성을 비교하는데 그 목적이 있으므로 시스템 특성(사양), 구현 요소 등에 대해서는 고려하지 않는다.

질의처리 시간은 변환된 RDF 그래프 상에서 질의 결과를 생성하기 위해 탐색하게 되는 아크의 개수에 의해 결정된다. 이는 RDF의 특성에 기인한 것으로 질의 효율성은 질의 값을 검색하기 위해 경유하는 아크의 수가 적을수록 효율성이 높다. 따라서, 질의 효율성은 질의처리를 위해 경유하는 아크의 수에 반비례하므로 경유한 아크 수의 역으로 표현된다.

특정 질의 결과를 생성하기 위해 변환 결과인 RDF 인스턴스 (그래프)상에서 거치게 되는 아크의 수를  $n(Arc)$ 라고 할 때, 질의 효율성  $QueryEff(CM)$ 은 다음과 같다. 이 때, 각 아크를 탐색하는 단위 시간은 동일하다고 가정한다.

$$QueryEff(CM) = \frac{1}{\sum Arc_i} = \frac{1}{n(Arc)}$$

## 5.2 비교 평가 결과 및 분석

이 절에서는 5.1절에 기술한 평가 모델을 이용하여 실험한 결과 및 이에 대한 분석 결과에 대하여 기술한다. 먼저 그림 10(a)는 [5]에서 제안한 변환 모델과 NOWL 모델의 의미 손실률 두 모델에 의해 변환된 RDF 인스턴스들이 각각 표현하는 의미 정보의 개수로 보여준다.

그림 10(a)에서, x-축은 토픽 맵이 지니는 의미의 개수를 의미한다. 그림 3 (3.1절)의 토픽 맵 예제를 [5]의 방법으로 변환하면 그림 5 (3.1절)와 같이 2개의 RDF 문장이 생성되거나 NOWL을 이용하여 변환할 경우 5개의



RDF 문장이 생성된다. 이를 평가 모델  $LossR(CM)$ 을 이용하여 의미 손실률을 산출하면 *NOWL*의 경우 의미의 손실이 없으므로  $LossR(NOWL) = 0.0$ 이고, [5] 변환 모델의 경우 5개의 의미 중 2개만이 생성됨에 따라 손실률은  $LossR([2]) = 0.6$ 이다. 그림 10(a)는 토픽맵의 크기가 증가하는 것을 가정하고 동일비율로 의미의 변환 갯수가 차이가 남을 표현한 것이다.

그림 10(b)는 *NOWL*과 [8]에서 제안한 변환 모델을 이용하여 변환된 RDF 인스턴스들의 구조적 복잡도에 대한 평가 결과를 보여준다. 그림 10(b)에서, [8]에서 제안한 모델은 *PivotCM*으로 정의함에 따라 변환된 RDF 인스턴스의 구조적 복잡도는 1이 된다. 따라서 비교 대상인 *NOWL* 변환 모델에 의해 생성된 RDF 인스턴스의 구조적 복잡도만이 변화한다. 그림에서 알 수 있듯이, *NOWL* 모델에 의해 생성된 RDF 인스턴스의 복잡도의 범위는 [0.76, 0.83]이다. 결론적으로, *NOWL* 모델이 RDF의 구조적 복잡도 측면에서 평균적으로 약 21% 정도 복잡도를 줄일 수 있다.

그림 10(c)는, 질의 효율성, 즉 질의처리 시간인 질의 결과 생성을 위해 경유하는 아크 수의 역을 보여준다. 질의 효율성의 비교 대상으로, [5]에서 제안한 변환 모델은 많은 의미 정보의 손실을 생성하기 때문에 의미가 없으므로 [8]의 변환 모델과의 비교 평가에 초점을 둔다. 질의 효율성, 즉 질의처리 시간 문제에 대한 평가를 위해 다음과 같은 질의를 가정해 보자.

**질의 예제:** "Who is a composer of the Tosca?"

위 질의에 대해 *NOWL* 모델에 의해 변환된 RDF 그래프 (인스턴스)의 경우, Tosca → Compose.Composer → Puccini 와 같은 순으로 노드를 검색함으로써 질의 결과를 생성하게 된다. 반면 [8]의 변환 방법에 의해 생성된 RDF 그래프 (그림 4 참조)의 경우에는 Tosca → Assocmember2 → Accoc:1 → Accosmember1 → Composer → Puccini와 같은 순으로 노드를 검색해야만 질의 결과를 생성할 수 있다. 이 예제에서 알 수 있듯이, *NOWL*의 경우 아크를 2회 경유함으로써

$QueryEfff(NOWL)=1/2=0.5$ 이다. 반면, [8]의 방법은 5회의 아크 경유를 통해서만 질의 결과를 생성함에 따라  $QueryEfff([3])=1/5=0.2$ 가 된다.

실험 결과에 따르면 *NOWL*에 의해 변환된 RDF 그래프가 [8] 방법에 의해 변환된 RDF 그래프 보다 평균적으로 약 48% 이상 높은 효율성을 지닌다. 이는 질의 처리 시간이 약 48% 이상 감소됨을 의미한다.

## 6. 결론 및 향후 연구

지금까지 시멘틱 웹의 실현을 위해 토픽맵과 RDF간 변환에 대한 다양한 연구가 진행되어 왔다. 그러나 기존의 주요 변환 모델들은 새로운 용어의 정의, 추가적인 노드의 정의, 의미의 구조적 왜곡, 의미 손실 등의 문제점을 지닌다. 이러한 문제점 중에서 가장 중요하게 해결되어야 할 사항은 의미 정보의 손실을 줄이고 원형의 의미 구조를 유지하는 것이다. 이를 위해 이 논문에서는 새로운 변환 방법인 *NOWL*을 제안하였다. *NOWL*은 기존 방법에서 간과한 *role*을 변환의 주요 대상으로 정의하고 *role*이 내포하는 의미 정보를 추출함으로써 의미 손실 문제를 해결하였다. 또한 제안한 방법은 불필요한 공백 노드 및 새로운 어휘를 사용하지 않음으로써 구조적인 왜곡 문제를 해결하고 친밀성을 유지하였다.

5장의 평가 결과를 통해 알 수 있듯이, 이 논문에서 제안하는 *NOWL* 모델은 의미 손실률, 구조 복잡도 및 질의 효율성 측면에서 기존 모델보다 나은 성능을 보임을 알 수 있다. 평가의 정당성을 위해 기존 연구 [6]에서 이용했던 평가 항목을 기준으로 평가 모델을 정의하였다. 의미 손실률에 있어서, *NOWL* 모델은 그림 3 (3.1절)에 주어진 토픽맵 예제로부터 5개의 RDF 문장 (의미 정보)을 변환하여 생성하였다. 반면, [5]의 방법으로는 2개의 RDF문이 추출됨으로써 *NOWL* 모델의 의미 손실률이 낮음을 알 수 있다. 구조 복잡도 측면에서는 *NOWL* 모델은 *subject*, *object*에 해당하는 노드가 4개 추출되나 [8]의 방법으로는 8개의 노드를 추출함으로써 같은 의미를 *NOWL*모델이 좀더 적은 노드로 변환해

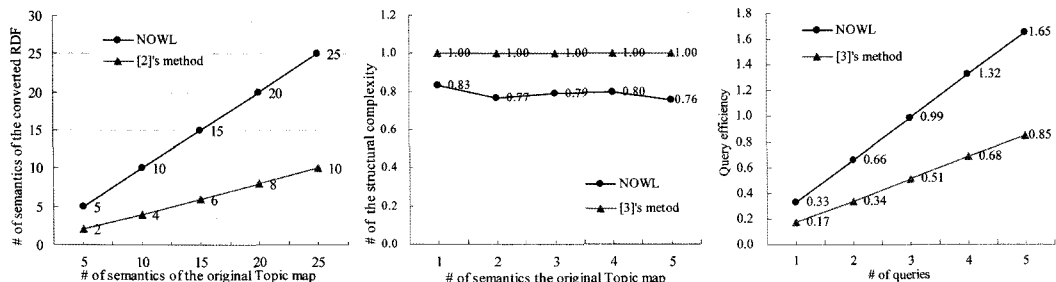


그림 10 평가 결과: 순차적으로 의미 손실률(a), 구조적 복잡도(b), 질의 효율성(c)

념을 알 수 있었다. 또한 주어진 질의에 대한 처리에 있어서, 기존 모델에 비해 낮은 질의 비용이 요구됨을 알 수 있다.

최근 웹 표준화 기구인 W3C에서도 토픽맵과 RDF간의 상호운용을 위한 작업반이 구성될 정도로 시멘틱 웹을 구현하기 위해서는 이 문제가 매우 중요하다[6]. 이 연구의 결과는 W3C 작업반에 제안하여 활용될 수 있을 것이며, 토픽맵을 통하여 정의된 웹 자원들은 NOWL 모델을 통하여 자동으로 RDF로 변환되어 활용될 수 있을 것이다.

향후 연구로는, 시멘틱 웹의 다른 구성 요소와의 관련성에 대한 연구가 요구된다. 이 연구에서는 RDF만을 대상으로 하였으나 RDF 스키마와 OWL도 추가로 고려하면 토픽맵이 가지고 있는 세부 정보들이 RDF 계열 기술(RDF family)로 변환될 수 있을 것이다. 또한 RDF를 토픽맵으로 변환하는 규칙도 정의가 요구된다. 이는 토픽맵과 RDF간의 완전한 상호운용성을 지원함으로써 실질적인 시멘틱 웹 실현에 기여할 것이다.

### 참 고 문 헌

- [1] Lassila, O. and Swick, R.R., "Resource Description Framework(RDF): Model and Syntax Specification," W3C, W3C Recommendation, February 22, 1999, <http://www.w3.org/TR/1999/REC-rdf-syntax-19990222>.
- [2] Biezunski, M., Bryan, M., and Newcomb, S., "ISO/IEC 13250, Topic Maps (Second Edition)," ISO/IEC JTC1, May 22, 2002, <http://www.y12.doe.gov/sgml/sc34/document/0322.htm>.
- [3] Brickley, D. and Guha, R.V., "RDF Vocabulary Description Language 1.0: RDF Schema," W3C, W3C Recommendation, February 10, 2004, <http://www.w3.org/TR/rdf-schema/>.
- [4] Garshol, L.M., "Living with topic map and RDF," deepX LTD., Proceeding of XML Europe 2003, London, England, 5-8 May, 2003.
- [5] Lacher, M.S. and Decker S., "On the Integration of Topic Maps and RDF Data," IOS Press, International Semantic Web Working Symposium (SWWS), California, USA, July 30-August 1, 2001, <http://www.semanticweb.org/SWWS/program/full/paper53.pdf>.
- [6] Pepper, S., Vitali, F., Garshol, L.M., Gessa, N., and Presutti, V., "A Survey of RDF/Topic Maps Interoperability Proposals," IDEAlliance Inc., XTech 2005 Conference, Amsterdam Rai Centre, Netherland, May 25-27, 2005.
- [7] Pepper, S. and Moore, M., "XML Topic Map (XTM) 1.0, TopicMap.Org," TopicMaps.Org, 2001, <http://www.topicmaps.org/xtm/1.0/>.
- [8] Moore, G., "RDF and TopicMaps: An Exercise in

Convergence," Proceeding of the XML Europe 2001 Conference, Berlin, Germany, May 21-25, 2001.

- [9] Ogievetsky, N., "XML Topic Map through RDF Glasses," MIT press, Journal of Markup Languages: Theory and Practice, Vol. 3. Issue 3, pp. 333-364, December, 2001.
- [10] Pepper, S., "Ten Theses on Topic Maps and RDF," Ontopia, August 2002, <http://www.ontopia.net/topicmaps/materials/rdf.html>.
- [11] Techquila, "Sample topic maps: The Clash," <http://www.techquila.com/tm-samples.html>.



신 신 애

1999년 고려대학교 컴퓨터과학기술대학원(석사). 2002년 고려대학교 컴퓨터학과(박사과정). 1993년~현재 한국전산원(책임연구원). 2005년~국내 SC32 메타데이터표준화(전문위원). 2002년~현재 TTA 메타데이터표준화 프로젝트그룹(위원) 2004년~현재 한국정보시스템감리사협회(이사). 관심분야는 데이터베이스, 시멘틱 웹, 메타데이터, 소프트웨어 공학, ITA/EA



정 동 원

1997년 군산대학교 컴퓨터학과(이학사). 1999년 충북대학교 전산과(이학석사). 2004년 고려대학교 컴퓨터학과(이학박사). 1999년~2000년 ICU 부설 한국정보통신교육원(전임강사). 2000년~2001년 지구넷 부설 연구소(선임연구원). 2002년~2005년 라인미디어 테크놀로지(연구원). 2004년 고려대학교 정보통신기술연구소(연구교수). 2005년 펜실베이니아 주립대학(PostDoc). 2002년~현재 TTA 표준화위원회-메타데이터표준화 프로젝트 그룹 PG406(특별위원). 2005년 SERA 2005 편집 위원. 2005년~현재 군산대학교 정보통계학과(교수). 관심분야는 데이터베이스, 이동 에이전트 시스템 및 보안, 유비쿼터스 컴퓨팅



백 두 권

1974년 고려대학교 수학과 학사. 1977년 고려대학교 대학원 산업공학과 석사. 1983년 Wayne State Univ. 전산학 석사. 1986년 Wayne State Univ. 전산학 박사. 1986년~현재 고려대학교 컴퓨터학과 교수. 1989년~현재 한국정보과학회 이사/학술위원장/편집위원장/평의원. 1991년~현재 한국시물레이션학회 이사/부회장/감사/회장/고문. 1992년~현재 ISO/IEC JTC1/SC32 국내위원회 위원장. 1996년~1997년 고려대학교 컴퓨터과학기술연구소 초대소장. 2001년~현재 도산아카데미연구원 원장. 2002년~2004년 고려대학교 정보통신대학 초대학장. 2004년~현재 한국정보처리학회 부회장. 관심분야는 메타데이터, 소프트웨어 모델링, 모델링/시물레이션, 정보기술표준 등